Pool of Tasks Runs (NOAH, on lomax, 1day)

The expected improvement in a parallel run can be estimated as follows. There are four main computational time hogs in GLDAS. space interpolation (*ipolates*), temporal interpolation (*zterp*), LSS runs (*noah_main*), and output routines (*noah_out*). Assume that the contributions of each of these routines towards the total execution time of GLDAS are N_1, N_2, N_3, and N_4 respectively. Let b_i, $i = 1, 4$ be a boolean value indicating if each of these routines are parallelized or not. The serial part left in GLDAS can be estimated as:

$$T_{ser} = T_1 \times (1 - \sum_{i=1}^{4} b_i N_i) \tag{1}$$

Assuming a linear speedup on parallelization, the improved total execution time can be estimated as:

$$\begin{aligned} T_{par} &= T_{ser} + \sum_{i=1}^{4} \frac{T_1 b_i N_i}{N} + T_{comm} \\ &= T_1 \times (1 - \sum_{i=1}^{4} b_i N_i + \sum_{i=1}^{4} \frac{b_i N_i}{N}) + T_{comm} \end{aligned} \tag{2}$$

The contribution of the tile loop towards the total execution time of GLDAS is approximately 11% and that of zterp is around 23% (from the sequential profiling results). Assuming zero communication overhead, the estimated times are shown in Figure 1. The improvements in *noah_main* and *zterp* loop timings are shown in Figures 2 and 3 respectively.

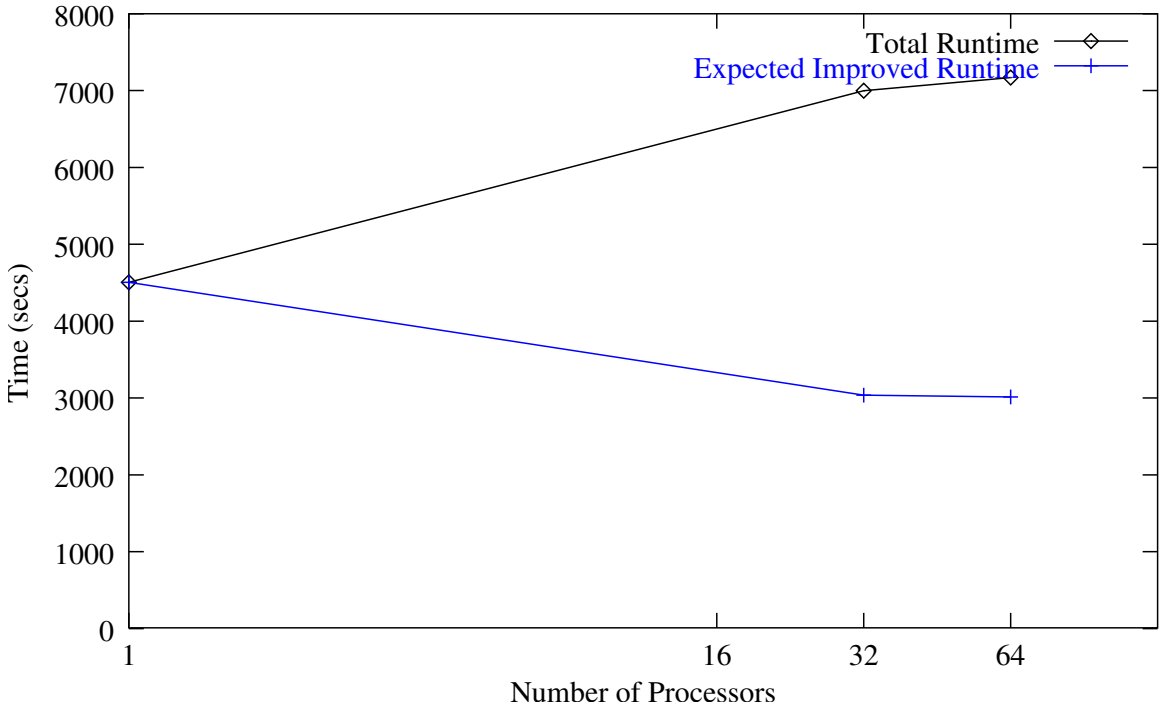

Figure 1: Total time for GLDAS runs with NOAH

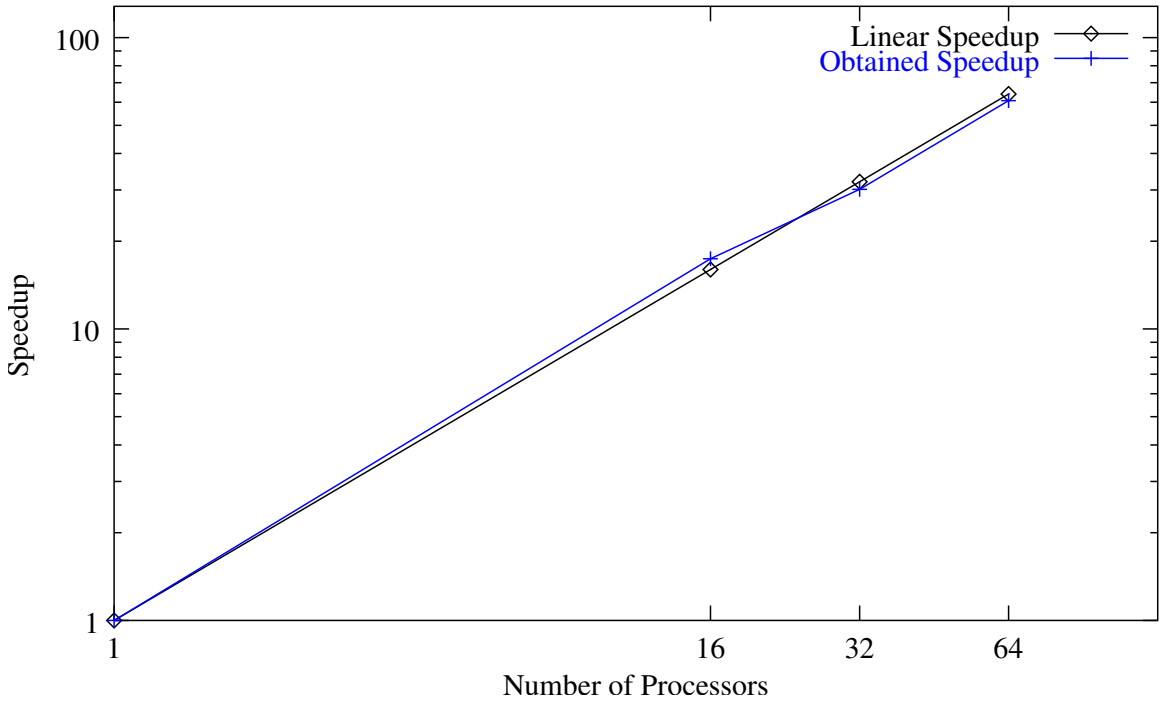

Figure 2: NOAH_MAIN loop timings for GLDAS runs

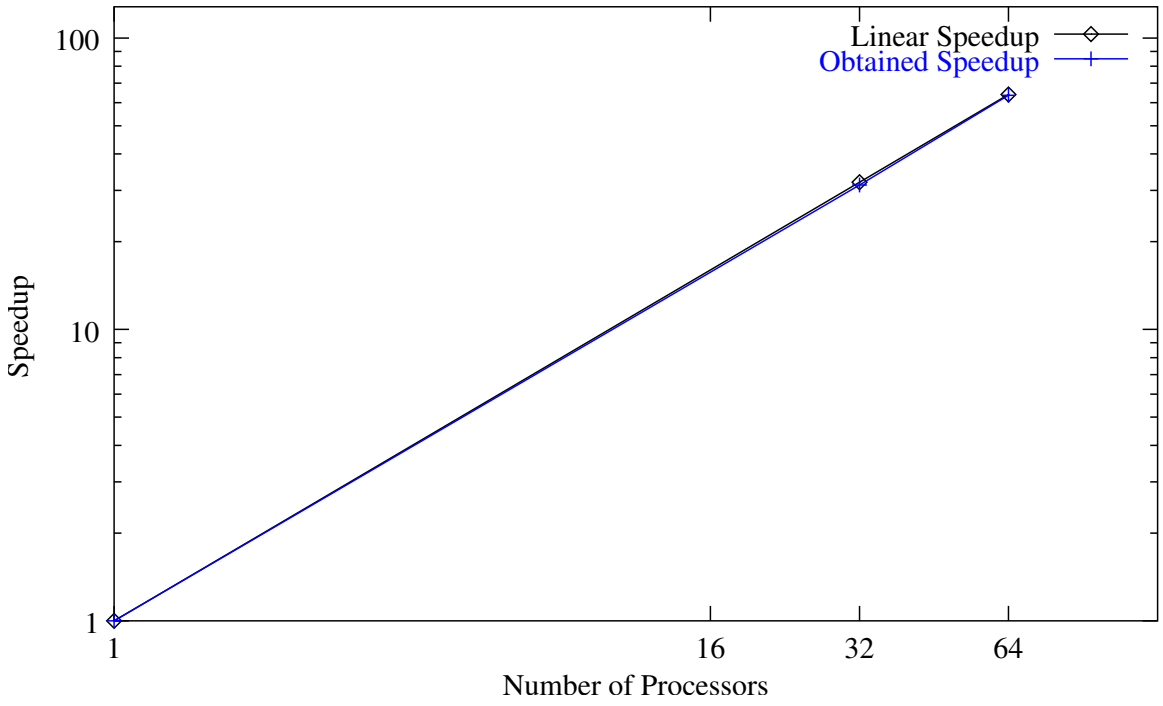

Figure 3: zterp loop timings for GLDAS runs